KIP 1. Logical Network to DISN Transport Backbone

This KIP is in essence multiple related KIPs. They reside primarily on technological rather than organizational boundaries. An example is the boundary between the NIPRNET and DISN backbone. The NIPRNET is a connectionless best effort packet-switched network riding on top of the connection oriented cell-switched ATM DISN backbone. Although these technologies complement each other they also preset difficult integration challenges. This KIP applies to other networks riding the DISN, including SIPRNET, DRSN, DSN, and DVS-G. It also applies to application level networks such as DMS Message Transfer Agents and service intranets such as NMCI.

KIP 2. Space to Terrestrial Interface

Future SATCOM will have network, transport, and possibly application layer interfaces. It is important to immediately start defining these interfaces in a way that retains flexibility as is practical. This KIP should be standardized across constellations. Otherwise, ground users will be forced to carry multiple terminal types.

KIP 3. JTF to Coalition

For a number of reasons, the U.S. is increasingly inclined to flight regional conflicts as part of a coalition. This presents real challenge for deployed forces, who must integrate diverse multinational C4I assets in the field and in the middle of crisis.

Because of the ad hoc nature of coalitions, in many cases neither the U.S. nor the coalition partners could have anticipated the requirement to the inter-network. In fact, even our known allies are often confounded in their attempts to prepare for U.S. interoperability, in that U.S C4I interfaces are numerous, diverse, poorly specified, and rapidly changing. The purpose of this KIP is to lend some predictability and specificity to the interface between U.S. forces and our allies. As a result, both U.S. and allied system and network builders will be better able to prepare for interoperability in ad hoc coalitions. In other words, we can move away from bilateral technical negotiations with every potential ally towards convergence on a smaller number of standard inter-network interfaces.

KIP 4. JTF Components to JTF HQ's

This is one of the more obvious Key Interface Points, in that every JTF will clearly require inter-networking between JTF headquarters and each component (whether Service of Functional). Less obvious is the fact that the same interface specification can probably also serve for inter-component interfaces (e.g. ARFOR to MARFOR or NAVFOR to AFFOR).

By carefully designing and specifying this KIP, Service architects can converge on implementations that allow virtually any reasonable task organization. In other words, it won't matter which Service CJTF comes from, what the compositions of JTF components might be, or who equips the JTF headquarters. The inter-network interfaces will be largely the same.

KIP 5. STEP and TELEPORT (i.e., deployed interface to DISN)

DISA is exploring ATM technology as a means for dynamically multiplexing STACOM traffic. But the Services are not all converging on the same solution as they plan for develop systems for the deployed end of STACOM links. Teleport KIPs are a vehicle for driving convergence on whatever technology emerges during joint development. The DoD Teleport system will involve migrating a collection of existing telecommunications hub points to configurations providing higher throughput and enhanced capabilities. These Teleport system locations will provide deployed forces with sufficient capabilities for multi-band and multimedia services between deployed locations throughout the world and DISN service delivery nodes for C4I needs. The Teleport system will facilitate interoperability between multiple satellite communications systems and deployed tactical networks.

KIP 6. Joint Interconnection Service

The JIS KIP is limited to a relatively small number of sites (approx 10 and a larger number of Internet Access Points (IAP) and is implemented primarily by a single organization (DISA). JIS sites are the first line in a defense in depth between the Internet and NIPRNET resources. A significant portion of NIPRNET traffic is internet bound. Therefore, many users and applications throughout the GIG will be impacted by details of JIS/IAP interfaces (e.g. TCP/IP ports & protocols screened.) IAP serves NIPRNET users on a single installation, unlike JIS sites, which serve the NIPRNET community at large. However, IAPs serve essentially the same purpose and present many of the same challenges as JIS sites. Therefore, it is appropriate to manage their NIPRNET/Internet interface in the same way.

KIP 7. DISN Service Delivery Point

This is the organizational, network management, and device ownership boundary between DISA and DISA-served sites. It often serves as a technology boundary, in that base/installation networks frequently employ different technologies than does the DISA backbone. A formally managed KIP will provide flexibility on both sides of the interface.

KIP 8. Secure Enclave Service Point

Ideally, this Key Interface Point (KIP) would look very much like the interface between any other enclave and its local DISN Service Delivery Point KIP. But encryption and multi-level networking technologies have not yet advanced to the point that secure enclaves can employ exactly the same technologies as unclassified enclaves. Also, technical mistakes potentially carry far greater consequences. Therefore, a higher degree of formality is appropriate in specifying this KIP.

The Secure Enclave SDP KIP extends from an installation's primary DISN service delivery point to a secure enclave's local area network. It includes (Type I) encryption interfaces, border gateway protocols, firewall specifications, IP Sec policies, intrusion detection, and other relevant technical details.

KIP 9. Applications to Database Server

This Key Interface Point (KIP) is important because it facilitates separation of mission specific applications from common user data infrastructure (in a three-tiered computing architecture). Thus, it will become easier for database developers to insulate themselves from requirements volatility at the user end. Application developers can focus on application logic and the user interface, rather than database design and development (or data collection and distribution). Increased standardization at this interface will also make it easier for developers to engineer interfaces relatively late in the development, in that most application servers will be fundamentally compatible with most database servers (even those for which an interface requirement was not initially anticipated).

KIP 10. Client Server

In all likelihood, this Key Interface Point will probably be a family of interface specifications rather than one KIP specification. For example, many applications can simply employ HTTP or HTTPS at the client interface. But other applications will have requirements that cannot be through a simple web interface.

Although application requirements will vary, it is desirable to converge on as few interface types as practical.

KIP 11. Applications to COE/CCP

This Key Interface Point (KIP) is important both as a boundary definition and as an interface specification. The boundary is important because applications developers must not duplicate functionality resident in the common computing platform or environment (since doing so would inevitably introduce compatibility issues). The interface specification is important because application developers must know what application support services reside in the computing platform, and how to access them (e.g., APIs).

KIP 12. End System to PKI

Many applications have a need to encrypt and/or authenticate users, transactions, clients, and severs. Traditionally, each application developer has built has built custom solutions to these problems. As a result, users are forced to deal with multiple password prompts, access controls, user registration processes, and other application-specific security mechanisms. The DoD PKI offers an opportunity to converge applications onto common and compatible security solutions. But this is only practical if the interface between applications and the PKI is clearly defined and relatively stable.

KIP 13. Management systems to (Integrated) Management Systems

The GIG actually consists of multiple tiers, each having its own management structure. And at the top level these management systems must interface with integrated management systems. To the extent practical all lower tier management systems should use common techniques and technologies for this interface with higher-level management systems. The role of this key interface point is to serve as a vehicle for that standardization.

KIP 14. Management Systems to Managed Systems

Commercial industry has long recognized the desirability of standardizing the interface between management systems and managed systems. In fact, SNMP is a de facto standard component of that interface. In particular, SNMP has long been recognized as a little weak in the area of security. So to be effective this KIP specification must go beyond a short list of open standards.

KIP 15. IDM to Distribution Infrastructure

Many communications systems will ultimately carry the information distributed by IDM. This key interface point is a vehicle for standardizing the interface between IDM and those communications systems. The most important aspect of this KIP is the interface between IDM and network management systems. This allows IDM to, for example, sense network and link state (e.g., congestion) and respond accordingly.

KIP 16. Information Servers to IDM Infrastructure

IDM infrastructure is tasked with managing the distribution of information from numerous information servers. It would not be logical to negotiate custom interfaces for every information server. Therefore, this Key Interface Point (KIP) should be used to converge all information servers on a common way of interfacing to IDM.

KIP 17. Applications to Shared Data

This Key Interface Point (KIP) is important because it facilitates separation of mission specific applications from their underlying data. Thus, it will become easier to insulate database implementations from requirements volatility at user end. And application developers can focus on application logic and the user interface, rather than database design and development (or data collection and distribution) when existing databases meet the application’s requirements. Finally, increased standardization at this interface will make it easier to migrate application databases into the shared data infrastructure for joint use (and configuration management) when they prove broadly useful (even those for which a joint interface requirement was not initially anticipated).